# Sisteme de Prelucrare Grafică

## Curs nr. 11

## 5.2 Texturarea prin perturbarea normalei la suprafaţă (bump – mapping)

În calculul iluminării unui punct intervine versorul normalei la suprafaţă în punctul respectiv, deci se pot simula diverse texturi prin perturbarea (modificarea) normalei. Procedeul este reprezentat în figura următoare:

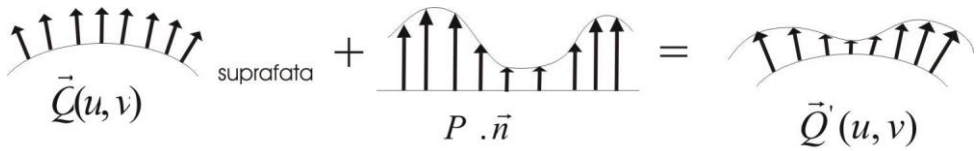


*Fig. 1*

unde:
- $P(u,v)$ - funcţia perturbatoare,
- $\vec{Q}(u,v)$ - funcţia vectorială care descrie parametrizarea suprafeţei.

Considerăm un punct al suprafeţei în care reprezentăm normala $\vec{N}$ şi vectorii tangenţi la suprafaţă $\vec{Q}_u$, $\vec{Q}_v$.

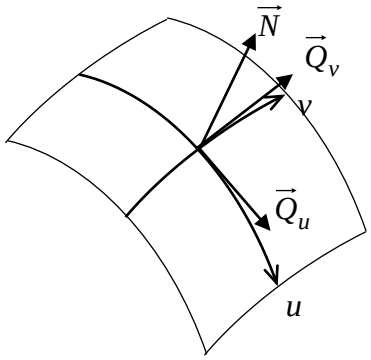


Fig. 2

$$\begin{cases} \vec{Q}_u = \dfrac{\partial \vec{Q}(u,v)}{\partial u} \\ \vec{Q}_v = \dfrac{\partial \vec{Q}(u,v)}{\partial v} \end{cases} \qquad (1)$$

$$\vec{N} = \vec{Q}_u \times \vec{Q}_v , \tag{2}$$

$$\vec{n} = \frac{\vec{N}}{\left|\vec{N}\right|} .$$

$$\vec{Q'}(u,v) = \vec{Q}(u,v) + P(u,v)\frac{\vec{N}}{\left|\vec{N}\right|} \tag{3}$$

=>

$$\vec{Q}'_u(u,v) = \vec{Q}_u(u,v) + P_u(u,v)\frac{\vec{N}}{\left|\vec{N}\right|} + P(u,v)\cdot\vec{n}_u$$

$$\vec{Q}'_v(u,v) = \vec{Q}_v(u,v) + P_v(u,v)\frac{\vec{N}}{\left|\vec{N}\right|} + P(u,v)\cdot\vec{n}_v$$

se neglijează

Noua normală este:

$$\vec{N}' = \vec{Q}'_u \times \vec{Q}'_v \quad =>$$

$$\vec{N}' = \vec{Q}_u \times \vec{Q}_v + P_u\frac{\vec{N}\times\vec{Q}_v}{\left|\vec{N}\right|} + P_v\frac{\vec{Q}_u\times\vec{N}}{\left|N\right|} + P_uP_v\frac{\vec{N}\times\vec{N}}{\left|N\right|^2} = 0$$

$$\boxed{\vec{N'} = \vec{N} + \underbrace{P_u\frac{\vec{N}\times\vec{Q}_v}{\left|\vec{N}\right|} + P_v\frac{\vec{Q}_u\times\vec{N}}{\left|N\right|}}_{\vec{D}}} \tag{4}$$

Ca funcţie perturbatoare se poate folosi orice funcţie derivabilă:

$$P_u = \frac{\partial P(u,v)}{\partial u} \ , \ P_v = \frac{\partial P(u,v)}{\partial v}$$

Efectul texturării prin această metodă e invariant faţă de translaţii însă nu e invariant faţă de scalare (pe măsură ce obiectul se măreşte, textura se netezeşte). Acest lucru se întâmplă deoarece funcţia de perturbare nu este scalată.

În practică se foloseşte (pentru a înlătura acest lucru)

$$\vec{D}' = \vec{D}\cdot\frac{\left|\vec{N}\right|}{\left|\vec{D}\right|}\cdot\sqrt{P_u^2 + P_v^2} \tag{5}$$

## 5.3 Aplicarea texturilor tridimensionale

Aplicarea texturilor tridimensionale se realizează mult mai simplu faţă de cele bidimensionale. Textura tridimensională poate fi privită ca un câmp de valori tridimensional continuu în întreg spaţiul ocupat de obiect.
Ignorând scalarea, atunci unui punct $(x_m, y_m, z_m)$ de pe obiect îi corespunde un parametru de textură dat de aplicaţia identitate $T(x_m, y_m, z_m)$.

$$(x_m, y_m, z_m) \longrightarrow T(x_m, y_m, z_m)$$

Deci culoarea unui punct al obiectului este dată de valoarea parametrului texturii (câmpului) din acel punct.

Acest mod de aplicare poate fi imaginat ca o sculptură a obiectului într-un bloc solid de material, de unde şi denumirea de texturare solidă (solid texturing).

*Avantaje*: - aplicarea directă a texturii pe suprafaţa obiectelor, fără distorsiuni.
*Dezavantaje*: - cantitatea enormă de memorie pentru stocarea unui câmp tridimensional complet de valori;
- nu se pot obţine orice fel de imagini, mai ales imagini nerepetitive, de genul fotografiilor.

## 5.4. Tehnici de anti-aliasing (filtrare) în texturare

Fenomenul de aliasing care apare în texturare se datorează eşantionării imaginii de textură. Transformarea din spaţiul ecran 3D în spaţiul texturii bidimensionale generează o suprafaţă în formă de patrulater corespunzătoare fiecărui pixel, suprafaţă numită *preimaginea pixelului.*

În continuare, pentru simplificare, presupunem că preimaginea pixelului este un pătrat în spaţiul texturii (fig. 3).

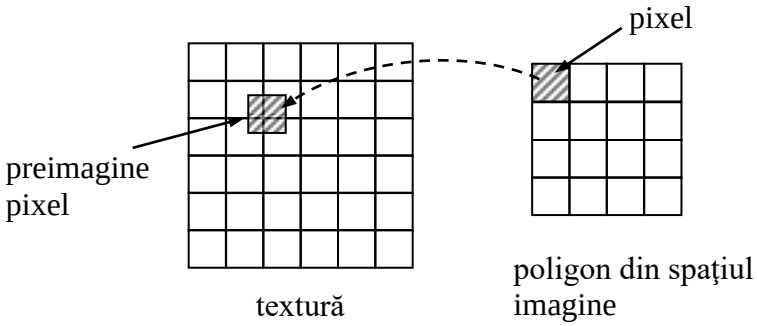


*Fig. 3* Preimaginea unui pixel în spaţiul texturii

Culoarea care se atribuie pixelului din figura de mai sus se obţine prin medierea ponderată a intensităţii culorilor celor patru texeli acoperiţi parţial de preimaginea pixelului, în felul acesta realizându-se filtrarea texturii.

Dacă nu se efectuează filtrarea texturii, atunci se atribuie pixelului culoarea texelului cel mai apropiat de centrul preimaginii. Problemele de aliasing, care apar în această situaţie, sunt foarte grave, mai ales pentru imaginile suprafeţelor privite în perspectivă, în care variază mult dimensiunea preimaginii pixelilor.

În continuare analizăm dependenţa preimaginii unui pixel de poziţia suprafeţei faţă de punctul de observare şi de direcţia de observare. Presupunem o textură pătrată cu $N \times N$ texeli şi o suprafaţă perpendiculară pe direcţia de observare, la o distanţă $D$, astfel încât imaginea ei pe ecran este un pătrat de $N \times N$ pixeli (fig. 4).

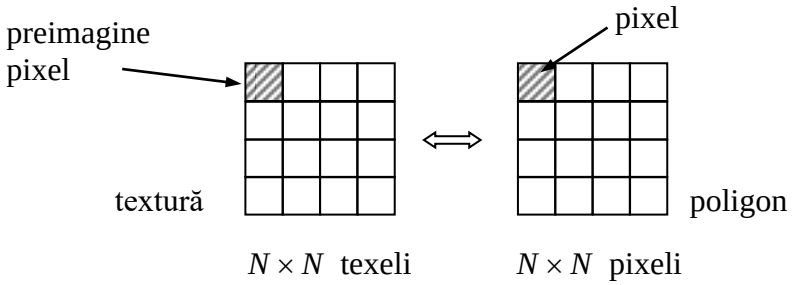


*Fig. 4* Pixel cu preimaginea cu arie egală cu a unui texel

Dacă suprafaţa se apropie de punctul de observare la $D/2$, atunci imaginea pe ecran are $2N \times 2N$ pixeli. Imaginea e acoperită de textură, deci $N \times N$ texeli corespund la $2N \times 2N$ pixeli, deci 1 pixel corespunde la $1/4$ texeli. Rezultă că preimaginea unui pixel este $1/4$ din aria unui texel (fig. 5).

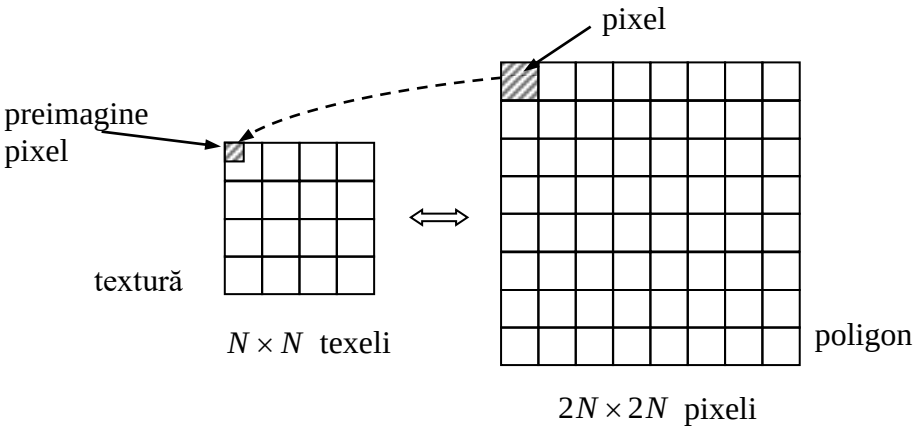


*Fig. 5* Pixel cu preimaginea cu arie mai mică decât aria texelului

Cu cât suprafaţa se apropie, cu atât scade dimensiunea preimaginii în spaţiul texturii, şi deci creşte frecvenţa de eşantionare a texturii. În această situaţie nu apare zgomot de aliasing, dar imaginea nu devine mai bună, deoarece se observă discontinuităţile de trecere de la un texel la altul (lucru datorat faptului că textura este o imagine discretă).

În cazul în care suprafaţa se depărtează la distanţa $2D$ avem situaţia din figura 6. Cei $N \times N$ texeli corespund la $N/2 \times N/2$ pixeli, deci 1 pixel corespunde la 4 texeli.

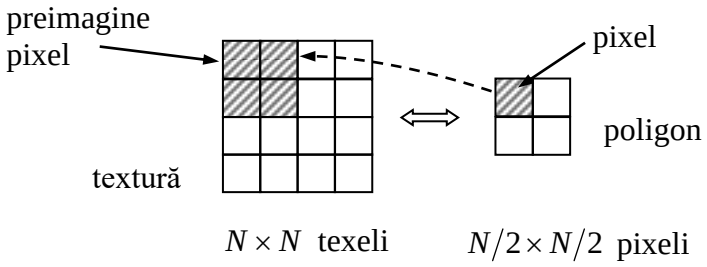


*Fig. 6* Pixel cu preimaginea mai mare decât un texel

Cu cât suprafaţa se depărtează, cu atât creşte dimensiunea preimaginii în planul texturii, deci scade frecvenţa de eşantionare. În acest caz apare zgomot de aliasing dacă nu se face filtrarea texturii. Filtrarea se efectuează considerând media ponderată a intensităţilor tuturor texelilor acoperiţi de preimaginea pixelului.

**Prefiltrarea texturilor**

Implementarea filtrării măreşte timpul de execuţie, care depinde de aria preimaginii. Procedeul de filtrare se poate accelera construind imagini prefiltrate ale texturii. De exemplu, în fig. 6 în loc să se efectueze medierea pe suprafaţa mai multor texeli, se poate folosi o imagine de textură cu rezoluţia de 2 ori mai mică, în care culoarea fiecărui pixel reprezintă media culorii texelilor din textură.

Această tehnică de prefiltrare se numeşte *tehnica mip-map* (mip - multum in parvo - multe lucruri într-un spaţiu restrâns). În loc să fie folosită o singură imagine de textură se foloseşte o succesiune de imagini ale aceleiaşi texturi, toate derivate din imaginea iniţială prin medierea culorilor şi reducerea rezoluţiei. Fiecare imagine are o rezoluţie egală cu jumătate din rezoluţia imaginii precedente, şi fiecare texel este media a 4 texeli din imaginea precedentă. În procesul de aplicaţie a texturii se alege textura cu aria texelului cea mai apropiată de aria preimaginii unui pixel. În OpenGL este implementată această tehnică.